

VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS

Absender: INTERNATIONALE RECHERCHENBEHÖRDE

An:

siehe Formular PCT/ISA/220

PCT

SCHRIFTLICHER BESCHIED DER INTERNATIONALEN RECHERCHENBEHÖRDE (Regel 43bis.1 PCT)

Absendedatum

(Tag/Monat/Jahr) siehe Formular PCT/ISA/210 (Blatt 2)

Aktenzeichen des Anmelders oder Anwalts
siehe Formular PCT/ISA/220

WEITERES VORGEHEN
siehe Punkt 2 unten

Internationales Aktenzeichen
PCT/EP2004/012687

Internationales Anmeldedatum (Tag/Monat/Jahr)
10.11.2004

Prioritätsdatum (Tag/Monat/Jahr)
06.12.2003

Internationale Patentklassifikation (IPK) oder nationale Klassifikation und IPK
G06F9/46

Anmelder
DAIMLERCHRYSLER AG

1. Dieser Bescheid enthält Angaben zu folgenden Punkten:

- ☒ Feld Nr. I Grundlage des Bescheids
- ☐ Feld Nr. II Priorität
- ☐ Feld Nr. III Keine Erstellung eines Gutachtens über Neuheit, erfinderische Tätigkeit und gewerbliche Anwendbarkeit
- ☐ Feld Nr. IV Mangelnde Einheitlichkeit der Erfindung
- ☒ Feld Nr. V Begründete Feststellung nach Regel 43bis.1(a)(i) hinsichtlich der Neuheit, der erfinderischen Tätigkeit und der gewerblichen Anwendbarkeit; Unterlagen und Erklärungen zur Stützung dieser Feststellung
- ☐ Feld Nr. VI Bestimmte angeführte Unterlagen
- ☐ Feld Nr. VII Bestimmte Mängel der internationalen Anmeldung
- ☐ Feld Nr. VIII Bestimmte Bemerkungen zur internationalen Anmeldung

2. WEITERES VORGEHEN

Wird ein Antrag auf internationale vorläufige Prüfung gestellt, so gilt dieser Bescheid als schriftlicher Bescheid der mit der internationalen vorläufigen Prüfung beauftragten Behörde ("IPEA"); dies trifft nicht zu, wenn der Anmelder eine andere Behörde als diese als IPEA wählt und die gewählte IPEA dem Internationale Büro nach Regel 66.1bis b) mitgeteilt hat, daß schriftliche Bescheide dieser Internationalen Recherchenbehörde nicht anerkannt werden.

Wenn dieser Bescheid wie oben vorgesehen als schriftlicher Bescheid der IPEA gilt, so wird der Anmelder aufgefordert, bei der IPEA vor Ablauf von 3 Monaten ab dem Tag, an dem das Formblatt PCT/ISA/220 abgesandt wurde oder vor Ablauf von 22 Monaten ab dem Prioritätsdatum, je nachdem, welche Frist später abläuft, eine schriftliche Stellungnahme und, wo dies angebracht ist, Änderungen einzureichen.

Weitere Optionen siehe Formblatt PCT/ISA/220.

3. Nähere Einzelheiten siehe die Anmerkungen zu Formblatt PCT/ISA/220.

Name und Postanschrift der mit der internationalen
Recherchenbehörde



Europäisches Patentamt
D-80298 München
Tel. +49 89 2399 - 0 Tx: 523656 epmu d
Fax: +49 89 2399 - 4465

Bevollmächtigter Bediensteter

Ebert, W

Tel. +49 89 2399-6016



Feld Nr. I Grundlage des Bescheids

1. Hinsichtlich der **Sprache** ist der Bescheid auf der Grundlage der internationalen Anmeldung in der Sprache erstellt worden, in der sie eingereicht wurde, sofern unter diesem Punkt nichts anderes angegeben ist.
 - ☐ Der Bescheid ist auf der Grundlage einer Übersetzung aus der Originalsprache in die folgende Sprache erstellt worden, bei der es sich um die Sprache der Übersetzung handelt, die für die Zwecke der internationalen Recherche eingereicht worden ist (gemäß Regeln 12.3 und 23.1 b)).
2. Hinsichtlich der **Nucleotid- und/oder Aminosäuresequenz**, die in der internationalen Anmeldung offenbart wurde und für die beanspruchte Erfindung erforderlich ist, ist der Bescheid auf folgender Grundlage erstellt worden:
 - a. Art des Materials
 - ☐ Sequenzprotokoll
 - ☐ Tabelle(n) zum Sequenzprotokoll
 - b. Form des Materials
 - ☐ in schriftlicher Form
 - ☐ in computerlesbarer Form
 - c. Zeitpunkt der Einreichung
 - ☐ in der eingereichten internationalen Anmeldung enthalten
 - ☐ zusammen mit der internationalen Anmeldung in computerlesbarer Form eingereicht
 - ☐ bei der Behörde nachträglich für die Zwecke der Recherche eingereicht
3. ☐ Wurden mehr als eine Version oder Kopie eines Sequenzprotokolls und/oder einer dazugehörigen Tabelle eingereicht, so sind zusätzlich die erforderlichen Erklärungen, daß die Information in den nachgereichten oder zusätzlichen Kopien mit der Information in der Anmeldung in der eingereichten Fassung übereinstimmt bzw. nicht über sie hinausgeht, vorgelegt worden.
4. Zusätzliche Bemerkungen:

**Feld Nr. V Begründete Feststellung nach Regel 43*bis*.1(a)(i) hinsichtlich der Neuheit, der
erfinderischen Tätigkeit und der gewerblichen Anwendbarkeit; Unterlagen und Erklärungen zur
Stützung dieser Feststellung**

1. Feststellung

Neuheit	Ja: Ansprüche 2, 4-6, 8 Nein: Ansprüche 1, 3, 7
Erfinderische Tätigkeit	Ja: Ansprüche Nein: Ansprüche 1-8
Gewerbliche Anwendbarkeit	Ja: Ansprüche: 1-8 Nein: Ansprüche:

2. Unterlagen und Erklärungen:

siehe Beiblatt

1 Dokumente

Es wird auf die folgenden Dokumente verwiesen:

D1: US-A-5 544 054 (KAYANO ET AL) 6. August 1996 (1996-08-06)

D2: US-A-5 155 851 (KRISHNAN ET AL) 13. Oktober 1992 (1992-10-13)

2 Einwände gemäss Artikel 33(2) PCT

- 2.1 Das Dokument D1 wird als nächstliegender Stand der Technik gegenüber dem Gegenstand des Anspruchs 1 angesehen. Es offenbart (die Verweise in Klammern beziehen sich auf dieses Dokument):

Ein Verfahren zum Laden eines Software-Moduls in eine Prozessoreinheit eines Steuergeräts in einem Verkehrsmittel, wobei

das Software-Modul in mehreren Steuergeräten lauffähig ist (Spalte 4, Zeilen 61-65) und die Steuergeräte über einen Datenbus Daten austauschen (Spalte 1, Zeilen 55-56),

die Auswahl auf welchem Steuergerät das Software-Modul geladen wird, in Abhängigkeit der Rechenkapazität der sich aktuell im Betrieb befindlichen Steuergeräte erfolgt (Spalte 1, Zeilen 44-49).

Der Gegenstand des Anspruchs 1 ist daher nicht neu.

- 2.2 Die abhängigen Ansprüche 3 und 7 enthalten keine Merkmale, die in Kombination mit den Merkmalen irgendeines Anspruchs, auf den sie sich beziehen, die Erfordernisse des PCT in bezug auf Neuheit erfüllen. Die Gründe dafür sind die folgenden:
- 2.3 Das Merkmal des Anspruchs 3, wonach das Steuergerät auf dem das Software-Modul läuft seine Rechenkapazität mit der Rechenkapazität der weiteren

Steuergeräte vergleicht und in Abhängigkeit des Vergleichs das Software-Modul beendet, wird in D1 offenbart (Spalte 11, Zeilen 14-16: jedem Prozessor wird die Rechenkapazität ("*load state*") aller anderen Prozessoren zur Verfügung gestellt; Spalte 3, Zeilen 45-54: Software-Module ("*control tasks*") werden entsprechend der Rechenkapazität der Prozessoren auf einem Steuergerät beendet ("*actuator B*"), auf ein anderes Steuergerät ("*actuator A*") transferiert und dort ausgeführt).

- 2.4 Das Merkmal des Anspruchs 7, das Software-Modul in den Speichermitteln der Steuergeräte zu speichern, wird in D1 offenbart (Spalte 3, Zeilen 51-54: "... *memorizing means* ...").

3 Einwände gemäss Artikel 33(3) PCT

- 3.1 Die abhängigen Ansprüche 2, 4 enthalten keine Merkmale, die in Kombination mit den Merkmalen irgendeines Anspruchs, auf den sie sich beziehen, die Erfordernisse des PCT in bezug auf erfinderische Tätigkeit erfüllen. Die Gründe dafür sind die folgenden:
- 3.2 Das Merkmal des Anspruchs 2, vor Ablauf des Software-Moduls zu ermitteln, welches der weiteren Steuergeräte freie Rechenkapazität zur Verfügung stellt und das Software-Modul auf einem solchen Steuergerät zu starten, wird in D1 offenbart (Spalte 7, Zeilen 40-45). Das Merkmal, das Steuergerät mit der maximal freien Rechenkapazität auszuwählen, ist eine häufig angewandte Load-Balancing-Strategie die der Fachmann ohne erfinderisches Zutun im System von D1 anwenden würde.
- 3.3 Das Merkmal des Anspruchs 4, die Rechenkapazität der Steuergeräte turnusmässig oder auf Anfrage zu ermitteln, sind fachübliche Möglichkeiten die in einem ähnliche Kontext bereits angewandt wurden, siehe z.B. Dokument D2 (Spalte 1, Zeilen 63-66: "... *predetermined time intervals* ...").
- 3.4 Dokument D1 offenbart das Merkmal von Anspruch 5, die Rechenkapazität eines Steuergeräts aus der Prozessorauslastung zu ermitteln (Spalte 3, Zeilen 2-4). Bei Vorhandensein verschiedener Prozessortypen ist es offensichtlich, dabei auch den Prozessortyp einzubeziehen.

- 3.5 Das Merkmal des Anspruchs 6, das Software-Modul auf dem Steuergerät mit der maximalen freien Rechenkapazität zu starten, ist eine fachübliche Grundstrategie in Load-Balancing-Systemen.
- 3.6 Das Merkmal des Anspruchs 8, wonach eine Kennung des Software-Moduls turnusmässig oder auf Anfrage auf den Datenbus gesendet wird, wobei die Kennung Informationen über Betriebszustand und Betriebs-Steuergerät des Software-Moduls enthält, ist eine Möglichkeit zur Verteilung solcher Informationen die im Rahmen des Allgemeinwissens des Fachmanns liegt.

DaimlerChrysler AG

August 16, 2005

Loading of software modules

The invention relates to a method for loading a software module into a processor unit in a controller in a means of transport, the software module being executable in a plurality of controllers and the controllers interchanging data via a data bus.

DE 196 31 309 A1 discloses a microprocessor arrangement for a vehicle control system having a plurality of microprocessor systems which are connected to one another by bus systems.

US 5 544 054 and US 5 155 851 respectively disclose a method for loading software modules into a processor unit in a controller. In this case, the selection regarding the controller on which the software module is loaded is made on the basis of the computation capacity of the controllers which are currently in operation.

EP 240 145 A2 discloses a system for selecting processors for handling tasks defined by software in a multiprocessor computer system. This method cannot readily be transferred to a means of transport on account of the real-time requirements and computation-time limitations.

The article "fine grained mobility in the emerald system, ACM transactions on computer systems", association for computing machinery, New York, US, 1988-02-00 discloses the forwarding of identifier information, such as the state of the host, in a computer system.

It is the object of the present invention to optimize the processor utilization level in controllers which are networked to one another.

The invention achieves this object by means of the features of Claims 1 and 9. Accordingly, a selection is made regarding the controller on which the software module is loaded, on the basis of the computation capacity of the controllers which are currently in operation. The selection method ensures that the software module currently has sufficient computation capacity available on the loaded controller for executing its processes and is not started on a controller on which there is currently insufficient computation capacity. The selection method allows targeted utilization of free computation capacities in a complex of controllers which can communicate with one another.

Preferably, the computation capacity of the controllers is ascertained in rotation or upon request. This has the advantage that it is known which controller currently has how much free computation capacity. This information can accordingly be used to control the loading of the software module onto a particular controller. The free computation capacity of a controller is dependent on the tasks which are currently to be handled by this controller. This is therefore subject to fluctuations and needs to be communicated to the other controllers.

Advantageously, the computation capacity of a controller is ascertained from the processor utilization level and the processor type, so that there is the assurance that even with different processor types the free computation capacity is determined correctly, in particular not only the processor utilization level is used.

Preferably, the software module is started on the controller with the maximum free computation capacity, so that controllers with little computation capacity are not burdened with executing the software module.

Preferably, the controller on which the software module is running compares its computation capacity with the computation capacity of the other controllers. On the basis of the comparison, the software module is terminated or continued by the controller. This has the advantage that the software module can be turned off in the event of processor utilization level alterations on the controller.

Advantageously, termination of the software module prompts ascertainment of which of the other controllers provides the maximum free computation capacity. In addition, the software module is started on this controller.

Daimler Chrysler AG

16 August 2005

Patent Claims

1. A method for loading a software module into a processor unit in a controller networked via a data bus in a means of transport, where the software module (7) is executable in a plurality of controllers (1, 3, 5) and the controllers (1, 3, 5) interchange data via the data bus (8), where the selection regarding the controller (1, 3, 5) on which the software module (7) is loaded is made on the basis of the computation capacity of the controllers (1, 3, 5) which are currently in operation, where

a check is performed to determine whether and on which controller (1, 3, 5) the software module (7) is running, this check is performed in rotation, that is to say in particular time periods, and each of the controllers (1, 3, 5) can turn off the software module (7) when the processor utilization level is high, where as soon as the software module (7) has been turned off the software module (7) is started again, and where the check to determine whether and on which controller (1, 3, 5) the software module (7) is running is performed by virtue of the software module (7) sending an appropriate identifier containing these data to the data bus (8) in rotation or upon request, and where it is established which of the controllers (1, 3, 5) involved in the data bus (8) has the maximum free computation capacity, that is to say the lowest processor utilization level in relation to the processor clock frequency, and where this information is obtained by virtue of the controllers (1, 3, 5) involved sending in rotation or by means of a request.

2. The method as claimed in claim 1, characterized in that before the software module (7) is executed it is ascertained which of the further controllers (1, 3, 5)

provides the maximum free computation capacity and the software module (7) is started on this controller (1, 3, 5).

3. The method as claimed in claims 1 and 2, characterized in that the controller (1; 3; 5) on which the software module (7) is running compares its computation capacity with the computation capacity of the other controllers (1; 3; 5) and terminates the software module (7) on the basis of the comparison.

4. The method as claimed in claims 1 to 4, characterized in that the computation capacity of a controller (1, 3, 5) is ascertained from the processor utilization level and the processor type.

5. The method as claimed in claims 1 to 5, characterized in that the software module (7) is started on the controller (1, 3, 5) having the maximum free computation capacity.

6. The method as claimed in claims 1 to 6, characterized in that the software module (7) is stored in the memory means in the controllers (1, 3, 5).

7. The method as claimed in claims 1 to 7, characterized in that an identifier for the software module (7) is sent to the data bus (8) in rotation or upon request, the identifier containing information about the operating state and the operating controller (1; 3; 5) of the software module (7).

8. A networked controller having software modules (2, 4, 6) stored in the microcontroller's memory, where the software modules (2, 4, 6) perform the primary tasks of the relevant controller (1, 3, 5), where a software module (7) with a subsidiary task can be additionally stored in a microcontroller's memory by the controllers (1, 3, 5), where the controllers (1, 3, 5) have process

cycles, where a process cycle is terminated after a particular time has elapsed, the data ascertained in the process are output onto the data bus (8), and the process cycle is started again, where the process cycle for the controllers (1, 3, 5) is determined by the software modules (2, 4, 6) for the primary task and/or the operating system and/or the bus protocol, and where when a process cycle or a process cycle time has elapsed data are sent to the data bus (8) which characterize their current processor utilization level and processor type used, with the controllers (1, 3, 5) using these data to ascertain the utilization level of the other controllers (1, 3, 5).